

Fig. 1

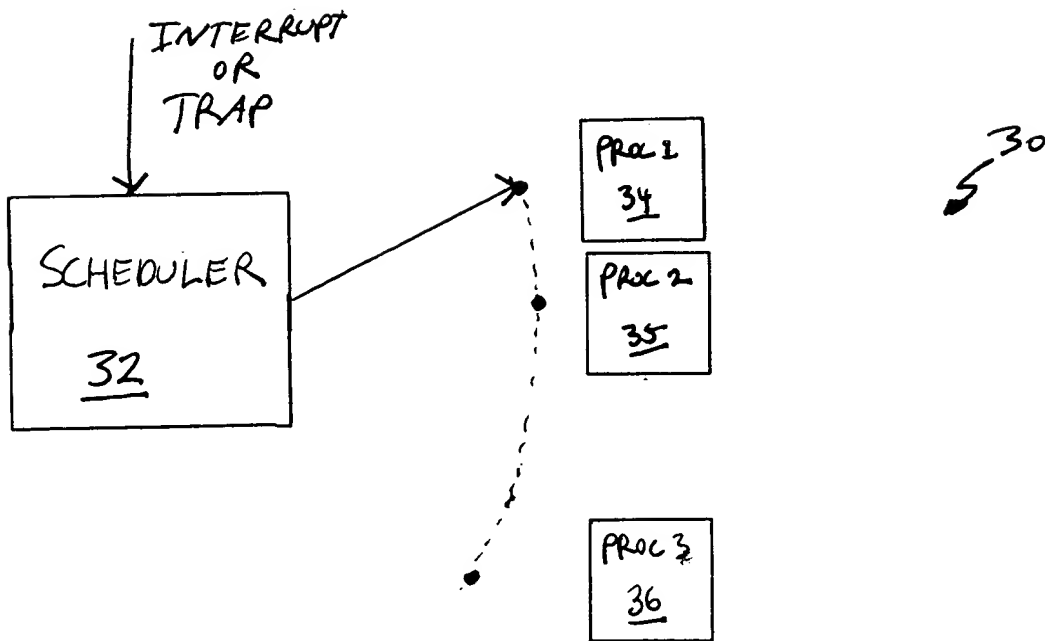
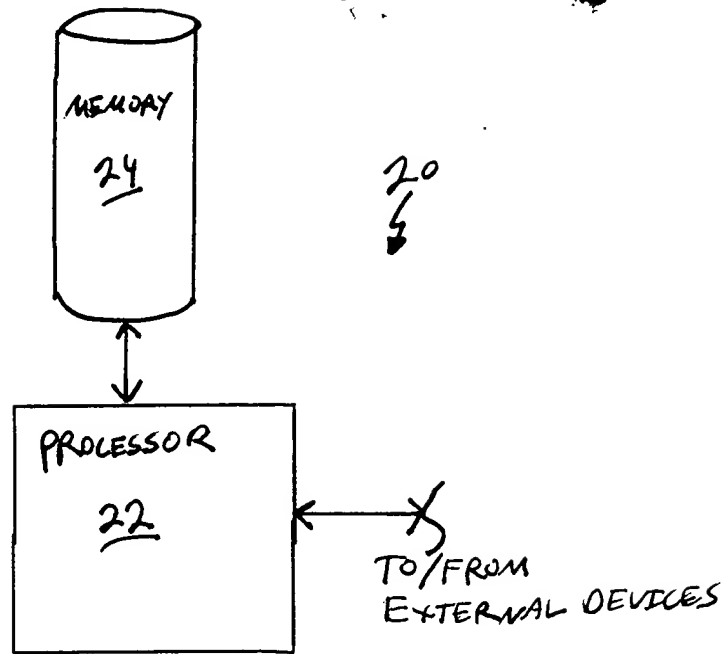


Fig. 2

40
41
42
43
44
45
46
47
48
49
50

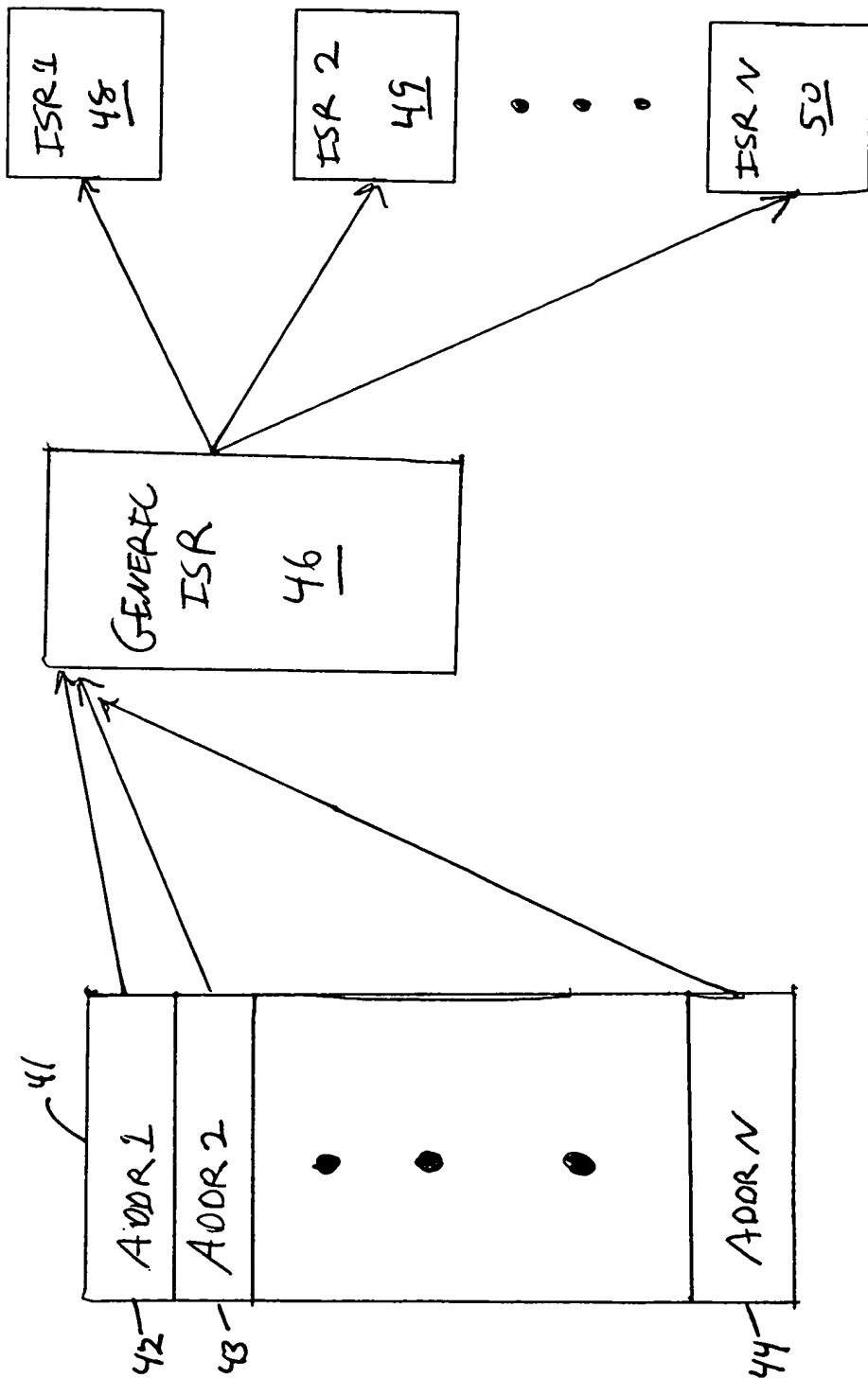


Fig. 3

ContextBlock**
_curcontext 62

ContextBlock*[] 64

0
1
2
3
MAXCONTEXTBLOCKS - 1

ContextBlock [] 66

0
1
2
3
MAXCONTEXTBLOCKS - 1

_contextblks

60 →

ProcEntry* 72

_curprocptr

ProcEntry [] 74

0
ContextBlock
1
ContextBlock
2
ContextBlock
3
ContextBlock
MAXPROCS - 1
ContextBlock

_proctable

Fig. 4

The diagram illustrates the scheduler context structure and its relationships. It shows a **ContextBlock**** array containing **_curcontext** (62), which points to a **ContextBlock*** array (64). This array contains elements 0, 1, 2, 3, and **MAXCONTEXTBLOCKS - 1**, with a pointer **_ctxtblkptrs** to the last element. The **ContextBlock*** array points to a **ContextBlock []** array (66), which contains **scheduler context**, 1, 2, 3, and **MAXCONTEXTBLOCKS - 1**. A dashed arrow from the **scheduler context** block points to a **scheduler** block (32) at the bottom. A handwritten note "80" with an arrow points to the **scheduler** block.

FIG. 5

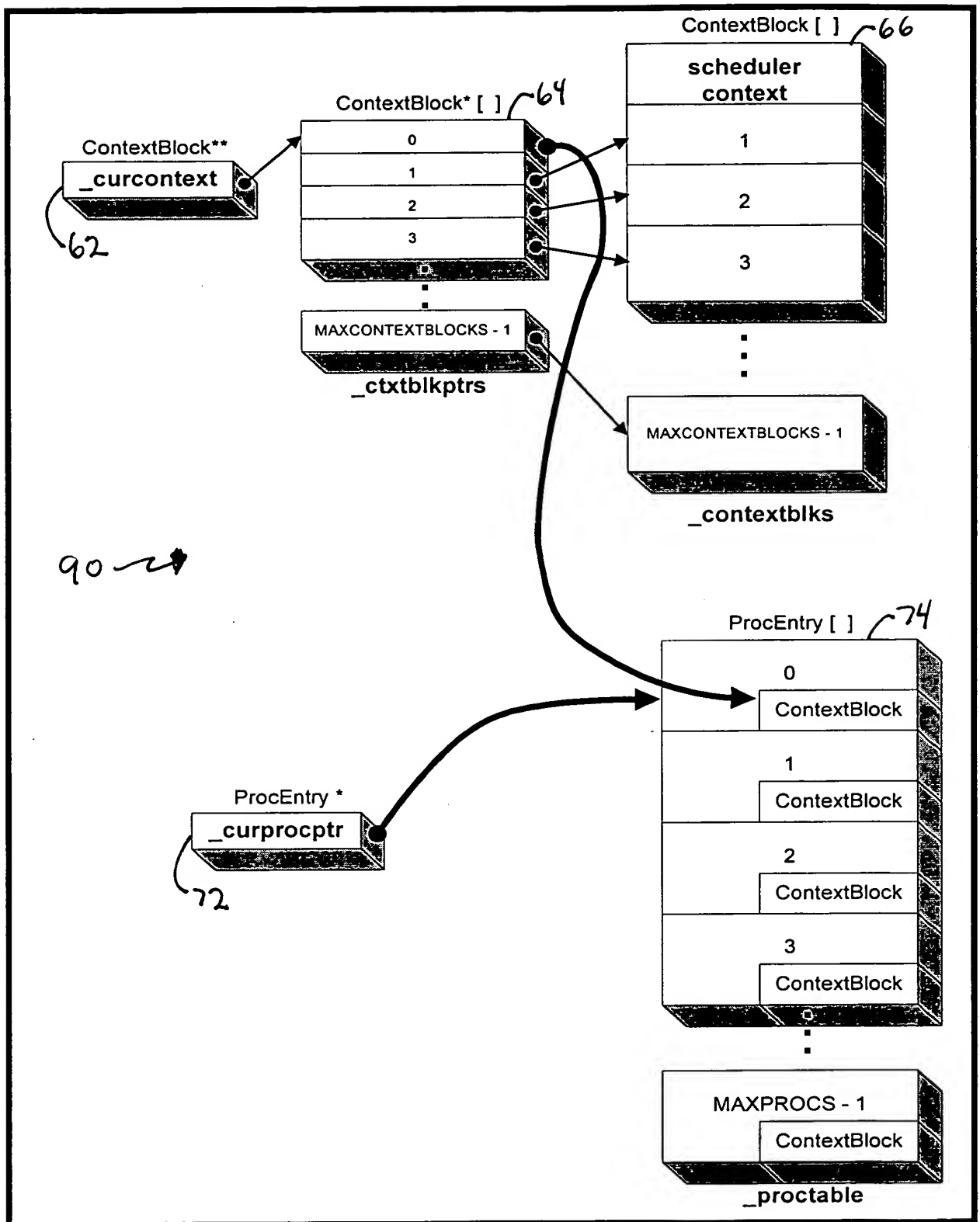


FIG. 6

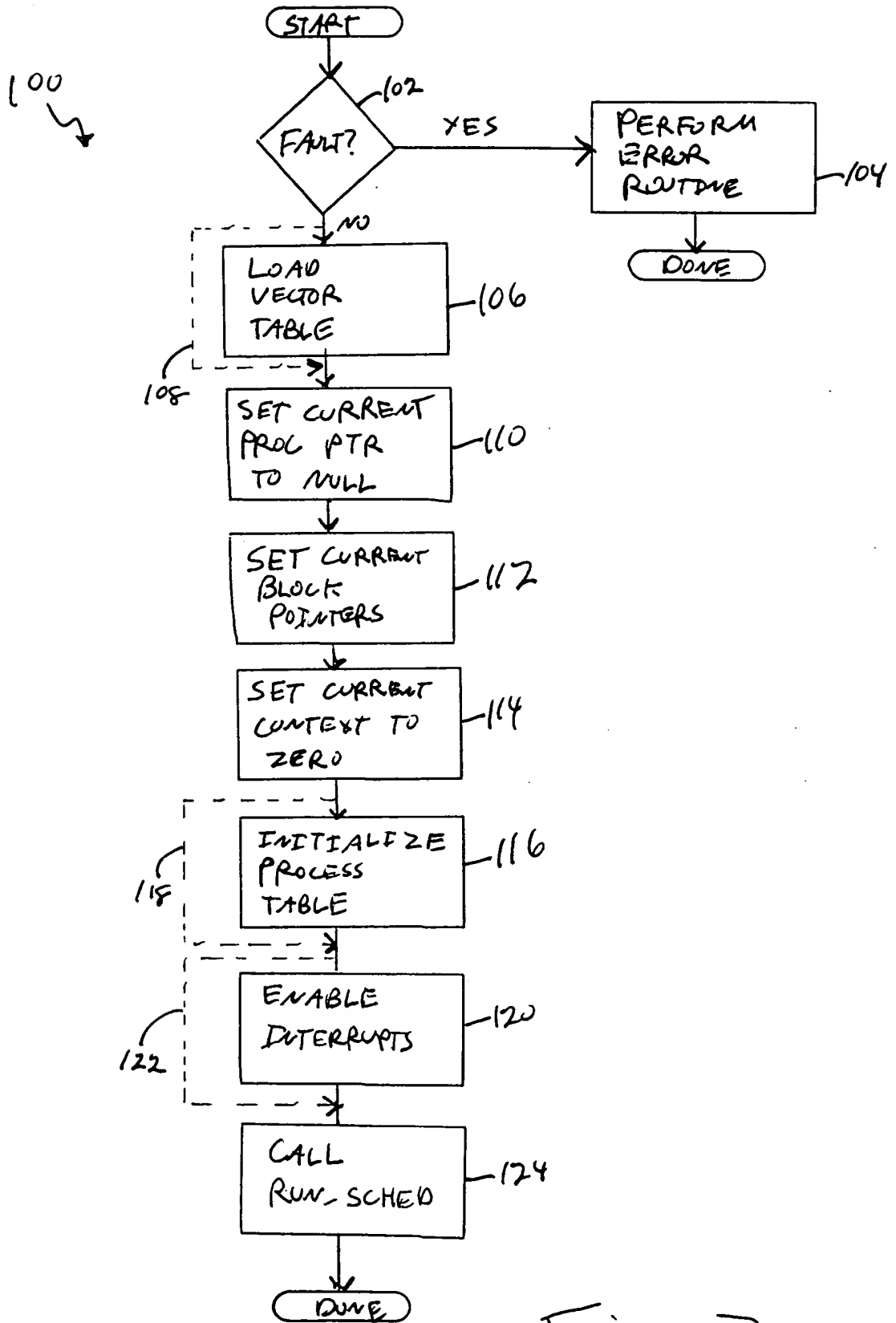


Fig. 7

DECEMBER 7

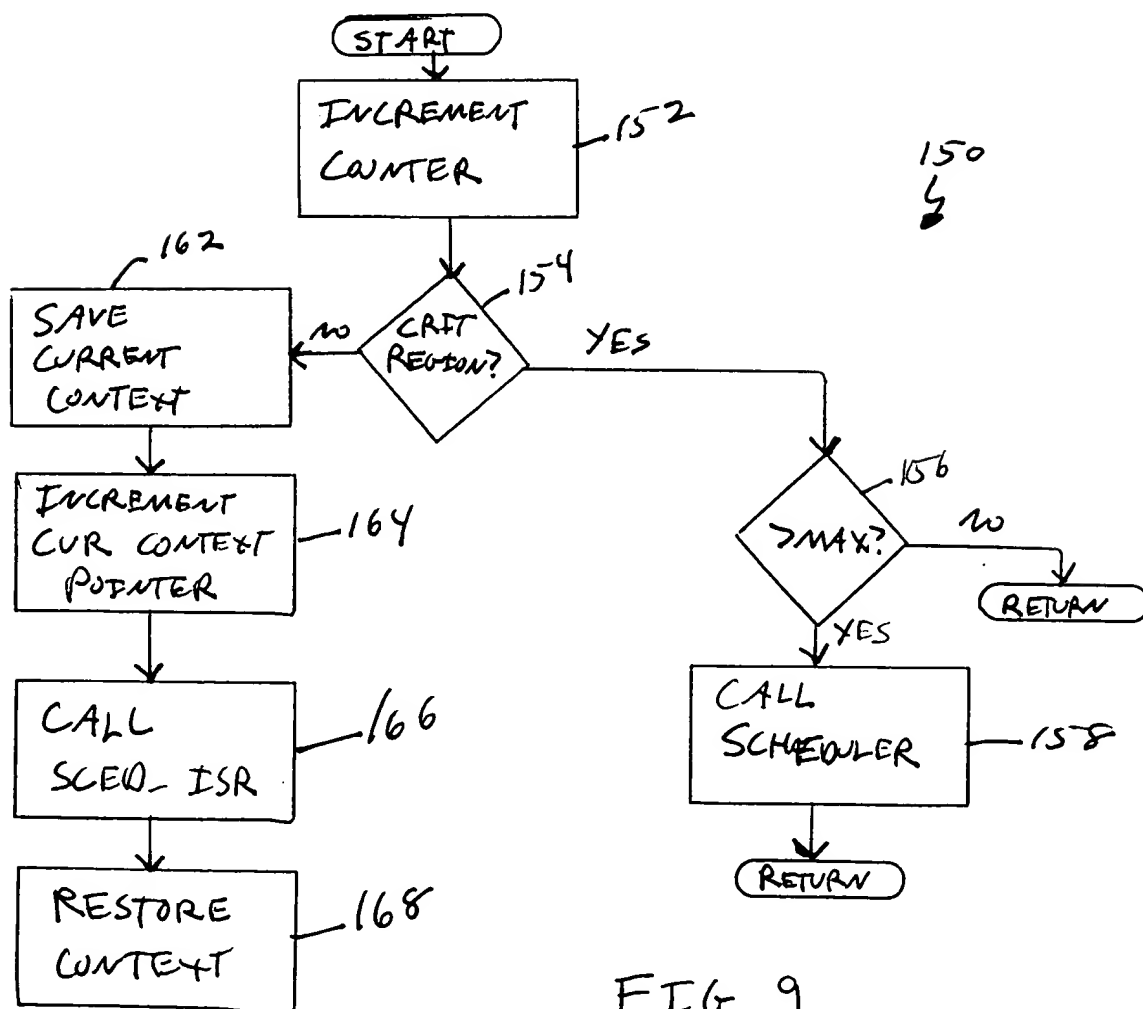
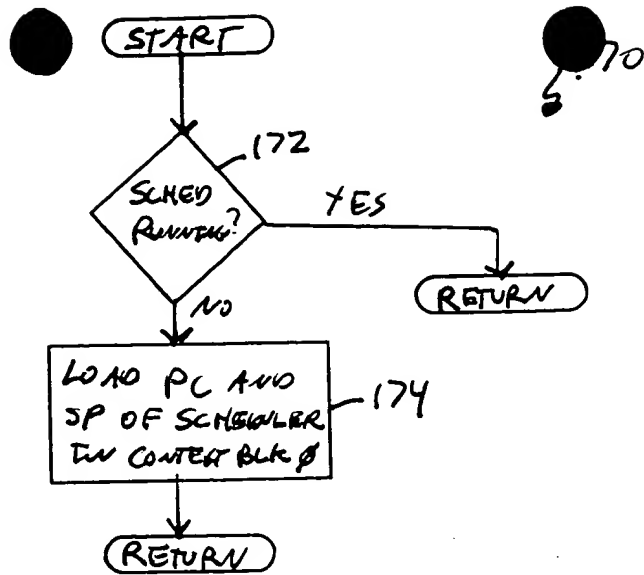


FIG. 9

[illegible]

F#6. 10

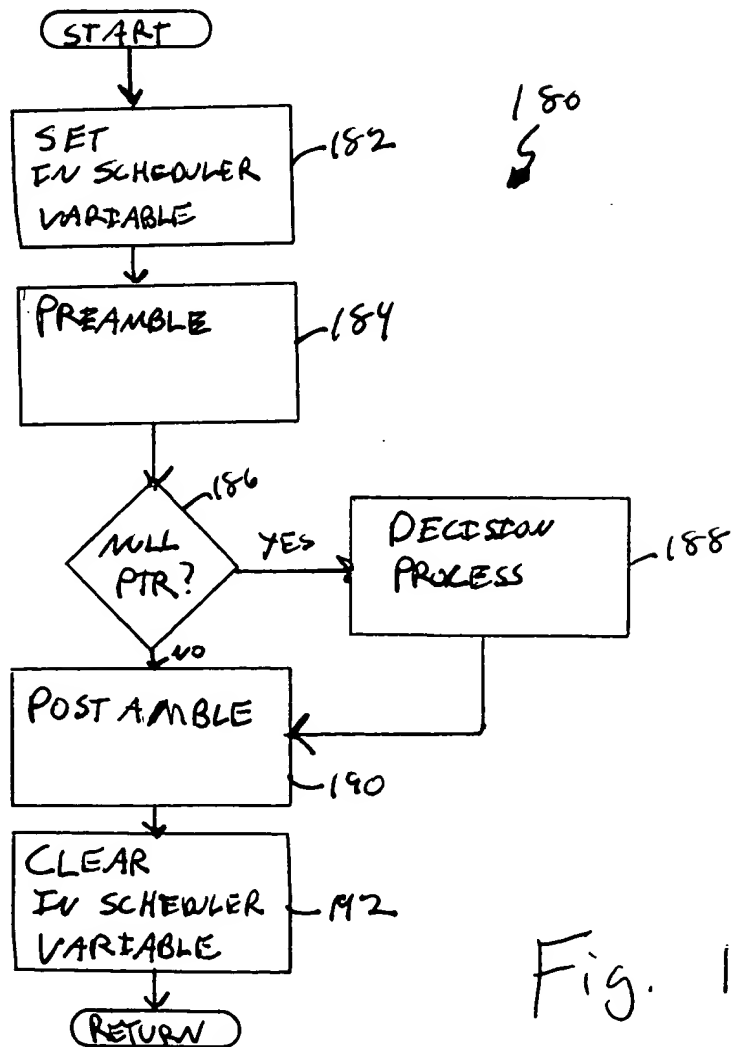


Fig. 11

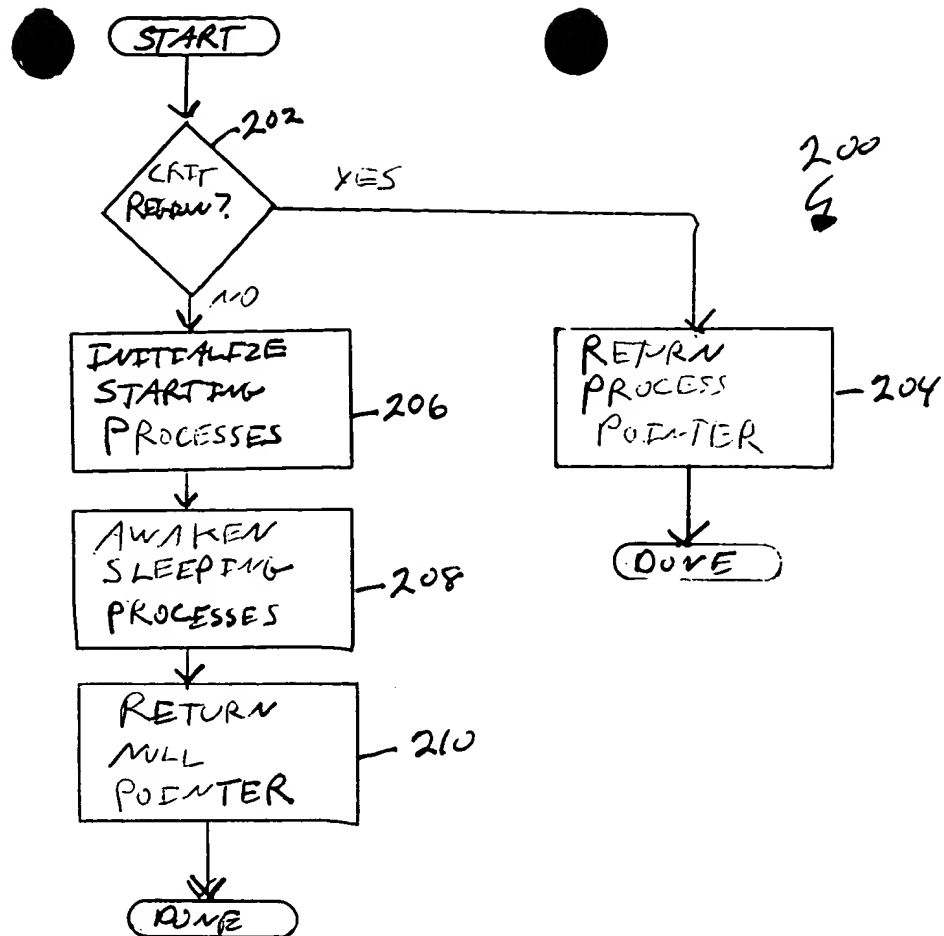


Fig. 12

SECRET

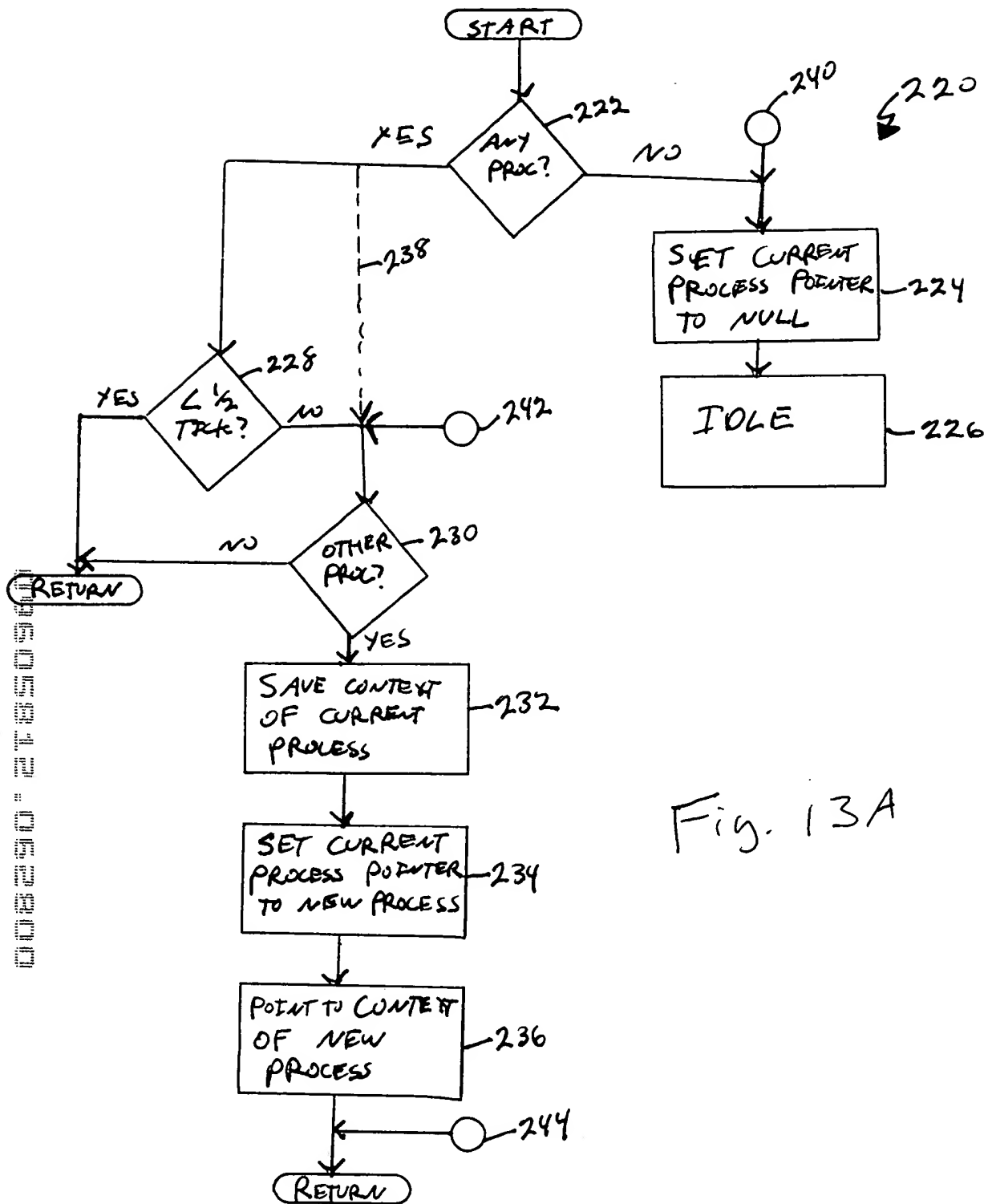


Fig. 13A

```

graph TD
    START([START]) --> 222{ANY PROC?}
    222 -- YES --> 252{RUN-NEXT SET?}
    222 -- NO --> 240(( ))
    240 --> 244(( ))
    252 -- YES --> 254[ CLEAR RUN-NEXT ]
    254 --> 242([RETURN])
    252 -- NO --> 242
    242 --> 244
    244 --> 256{SWAP IN BY REL?}
    256 -- NO --> 256_RETURN([RETURN])
    256 -- YES --> 258{7 1/2 TECH?}
    258 -- YES --> 258_RETURN([RETURN])
    258 -- NO --> 260[ SET RUN-NEXT ]
    260 --> 262_RETURN([RETURN])
    258 -- NO --> 262
    262 --> 258

```

FIG. 13B

264
5

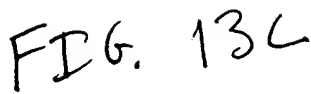


FIG. 13C

